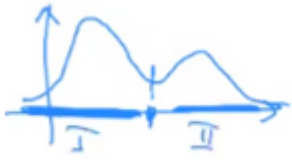
Непараметрические методы оценки плотности

Пусть есть $X_1, \dots X_n \sim P$, где P - абсолютно непрерывное распределение

Задача - оценить плотность $p : \mathbb{R} \to \mathbb{R}_+, \ \int p(x)dx = 1$

Зачем?

1. $p \ \Rightarrow \ \mathbb{E}, \mathrm{Var}$
2. кластеризация. Пример: в книге Wasserman “All of nonparam statistics” есть пример с оцениваем расстоянием до космических объектов. Есть сайт sdss.org. С помощью специальной методики оцениваются расстояние от земли до далеких небесных тел. После этого можно разделить тела на галактики. Это можно сделать, построив график плотности. С помощью такой методики удалось найти порядка тысячи галактик.



Непараметрическая оценка функции распредения (эмпирическая):

$$F_n(x) = \frac{1}{n} \sum \mathbb{I}\{x_i < x\}$$

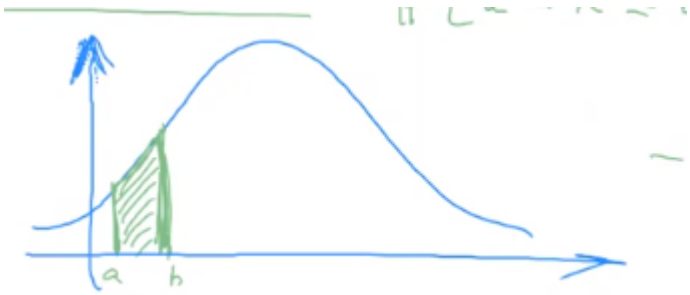
Она имеет много хороших свойств. Возникает идея посчитать плотность так:

$$\hat{p}_n(x) = \frac{d}{dx} \hat{F}_n(x)$$

Другая идея - гистограмма. В математике это понятие немного отличается от общепринятого представления. Считается вероятность того, что X принадлежит интервалу $(a, b]$:

$$P\{a < X \leq b\} = F(b) - F(a) = \int_a^b p(u)du$$

Хотим аппроксимировать интеграл:



Если аппроксимировать прямоугольником, то получаем:

$$P\{a < X \leq b\} = p(x)(b - a), \ x \in [a, b]$$

Так как

$$P\{x - \frac{h}{2} < x \leqslant x + \frac{h}{2}\} = p(x)h$$

то

$$p(x) = \frac{P\{x - \frac{h}{2} < x \leqslant x + \frac{h}{2}\}}{h} \approx \frac{\#\{i : x - \frac{h}{2} < x_i \leqslant x + \frac{h}{2}\}}{nh}$$

Значит, работаем по такой схеме: разбиваем гистограмму на интервалы бины и для каждого считаем статистикиЖ

- $B_1, \ldots, B_n$ - bins
- $|B_1| = \cdots = |B_m| = h$ - bandwidth
- $C_1, \ldots, C_n$ - центры

Они связаны так:

$$B_i = \left[C_i - \frac{h}{2}, C_i + \frac{h}{2}\right]$$

Будем оценивать плотность по формуле:

$$\hat{p}_n(x) = \frac{\#\{i : x_i \in B_j\}}{nh},\ x \in B_j$$

Как оценить качество?

Bias-variance decomposition tradeoff

Первое что приходит в ум - посчитать ошибку

$$\mathbb{E}[\hat{p}_n(x) - p(x))^2] = \mathrm{MSE}(\hat{p}_n(x))$$

Недостаток - зависит от x. Более устойчивая вещь это **mean integer square error**:

$$\mathrm{MISE}(\hat{p}_n(x)) := \int \mathrm{MSE}(\hat{p}_n(x))dx$$

Можно разложить MSE так:

$$\mathrm{MSE}(\hat{p}_n(x)) = (\mathrm{Bias}(\hat{p}_n(x)))^2 + \mathrm{Var}\,\hat{p}_n(x)$$
$$\mathrm{MISE}(\hat{p}_n(x)) = \int (\mathrm{Bias}(\hat{p}_n(x)))^2 + \int \mathrm{Var}\,\hat{p}_n(x)$$

В чем *компромисс*?

Теорема. Если

$$\int (p'(x))^2\,dx < \infty$$

то

$$\mathrm{MISE}(\hat{p}_n(x)) = \left(\frac{h^2}{12}\int \left(p'(x)\right)^2 dx + \frac{1}{nh}\right)(1 + \bar{o}(1)),\ n \to \infty$$

Проанализируем

- первое слагаемое растет как парабола (это bias)
- второе слагаемое убывает как гипербола (это дисперсия)

- есть минимум

Резюме: надо пользоваться MISE

Как выбирается параметр h в R?

Можно просто вычислить ноль производной функции выше:

$$\frac{d}{dh}AMISE(h) = \frac{d}{dh}(\frac{h^2}{12}\int(p'(x))^2\,dx + \frac{1}{nh})$$

$$h_{opt} = n^{-1/3}(\frac{6}{\int p'(x)^2\,dx})^{1/3}$$

AMISE - асимптотическая MISE

На самом деле можно было не вычислять производную.

1. $AMISE(h) = f(n)h^{k_1} + f_2(n)h^{k_2},\ k_1 = 2, k^2 = -1$

   Если продифференцировать то получим, что $h_{opt} = C \cdot (\frac{f_2(n)}{f_1(n)})^{\frac{1}{k_1+k_2}}$

   Но есть фокус: можно просто приравнять порядки:

   $$f(n)h^{k_1} = f_2(n)h^{k_2} \ \Rightarrow\ \tilde{h} = h_{opt}$$

   Это свойство верно и для дальнейших методов

   Но мы не можем вычислить эту оценку, т.к. не знаем интеграл. Оценка выше называется оракульной

2. Правило Скотта (Scott's rule):

   $$p(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\{-x^2/2\sigma^2\},\ N(0,\sigma^2)$$

   Отсюда $h_{opt} = n^{-1/3}(24\sqrt{\pi})^{1/3}\sigma \approx 3.5$. Идея:

   $$\hat{h}_{opt} = n^{-1/3}3.5\hat{\sigma}_n,\ \hat{\sigma}_n = \frac{1}{n}\sum(x_i - \bar{x})^2$$

3. Правило Фридмана-Дьякони (FD)

   $$\hat{h}_{opt} = n^{-1/3}\cdot 2\cdot \mathrm{IQR}(x_1,\ldots,x_n)\$\$$$

4. Метод Стерджеса

   Пусть у нас Биноминальное распределение:
   $\xi = \mu_1 + \ldots \mu_n,\ P(\xi = k) = C_n^k p^k(1-p)^{m-k}$. Нам нужно приблизить эту картинку к нормальному распределению. Если у нас есть n наблюдений, то оптимально взять $m = \log_2 n$.

> В языке R все формулы считаются по немного другим формулам

В R используется функция pretty, которая делает число "красивым". Число красивое, если первый знак после запятой это 2, 4, 6, 8, 5 или число вообще целое

Ядерная оценка плотности

density(X)

Kernel density esimate

Kernel $K : \mathbb{R} \to \mathbb{R}_+,\ \int K(x)\,dx = 1$, K - четная

- boxcar kernel: $1/2\mathbb{I}\{|x| < 1\}$
- triangle kernel: $(1 - |x|)\mathbb{I}\{|x| < 1\}$
- Epanechnikov kernel: $3/4(1 - x^2)\mathbb{I}\{|x| < 1\}$
- Gaussian kernel: $\frac{1}{\sqrt{2\pi}} \exp\{-x^2/2\}$

Kernel density estimate:

$$\hat{p}_n(x) = \frac{1}{nh} \sum K(\frac{x - X_i}{h})$$

Можно заметить, что

$$(\dots) \neq 0 \iff |\frac{x - X_i}{h}| < 1$$

Теорема. Если $\int (p''(x))^2\,dx < \infty$, то

$$\mathrm{MISE} = \Big(\frac{1}{4}h^4 \int (p''(x))^2\,dx \Big(\int x^2 K(x)\,dx \Big)^4 + \frac{1}{nh} \int K^2(x)\,dx \Big)(1 + o(1))$$

Приравниваем порядки: $h^4 = 1/nh \ \Rightarrow\ h_{opt} \sim n^{-1/5}$.

А с гистограммой мы получили $h_{opt} \sim n^{-1/3}$. Почему другой порядок?

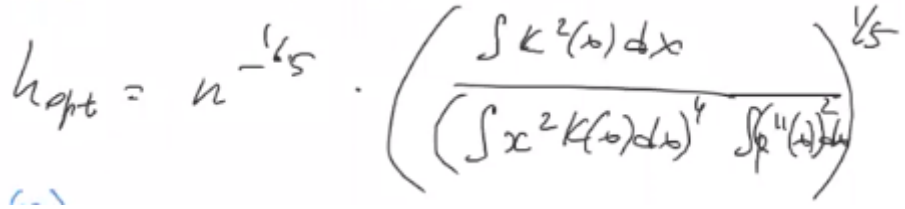
Теорема. Пусть $X_1, \dots X_n \sim \Phi_m = \{\text{abs. cont.},\ \int (p^{(m)}(x))^2\}$

Тогда

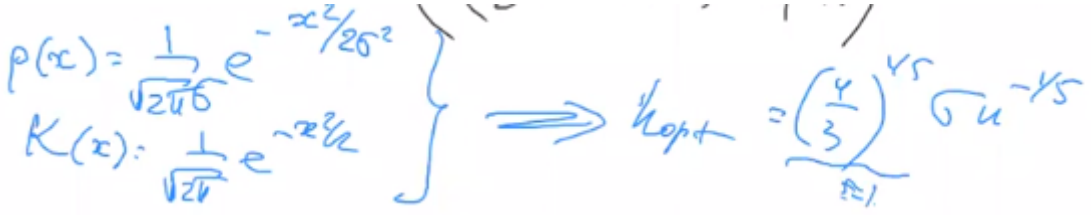
$$\sup_{p \in \Phi_m} \mathrm{MISE}(\hat{p}_n) \geqslant C \cdot n^{-\frac{2m}{2m+1}}$$

Соответственно, для Φ_1 у нас порядок $n^{-2/3}$, для Φ_2 порядок $n^{-4/5}$. То есть гистограмная оценка оптимальна в классе Φ_1, ядерная - в классе Φ_2.

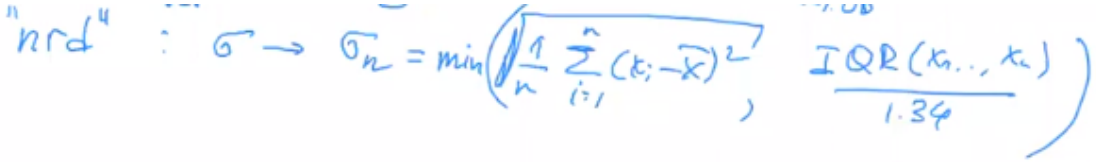
Итоговая оценка имеет вид



Можно предположить нормальность распределения и тогда получим оценку



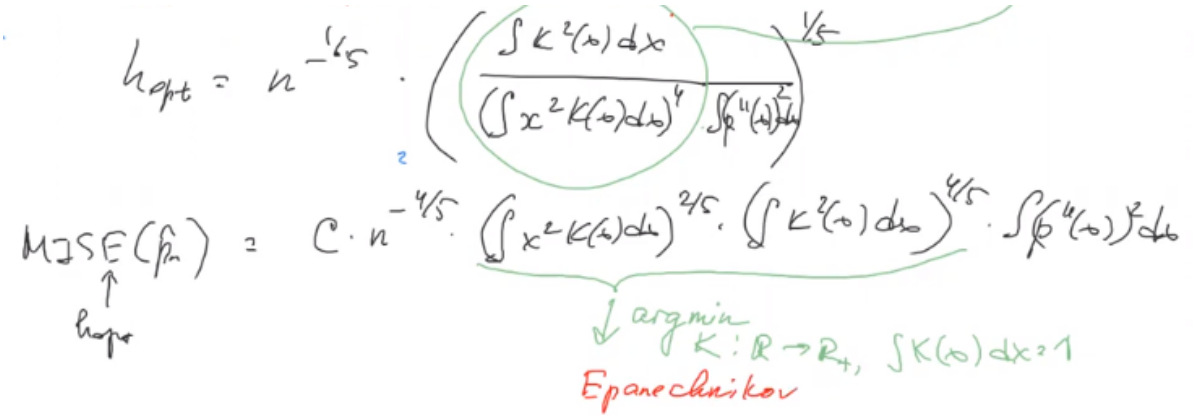
В R используется nrd для оценки дисперсии:



Есть еще аналогичное “nrd 0”. Вместо оценки $(4/5)^{1/5}$ берут 0.9 (чисто эмпирически).

Как выбрать ядро?

При фиксированном h мы получаем функционал от ядра, который можно оптимимзировать. И для этой задачи даже есть решение



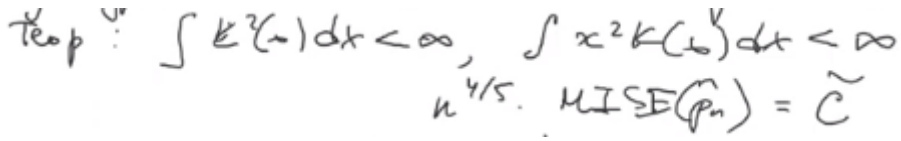
Ядро Епанечникова - самое лучшее. Но на самом деле выбор ядра не очень важно. Определим эффективность:

$$\text{eff}(K) = \frac{I(K_{ep})}{I(K)} \leqslant 1$$

Эта штука фигурирует в формуле. На самом деле, почти для всех ядер эффективность близка к 1, просто у Епанечникова она максимальная.

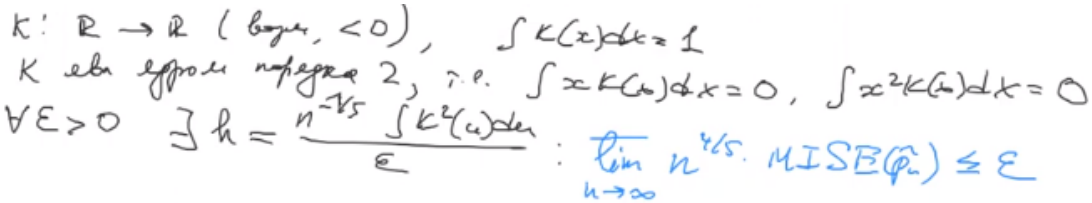
Почему ядро Епанечникова на самом деле не оптимально?

1. Выше была теорема:

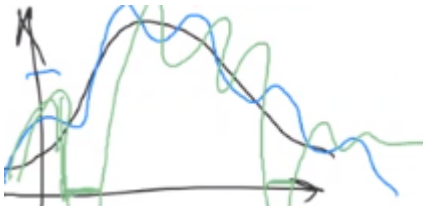
   У ядра Епанечникова константа максимальна

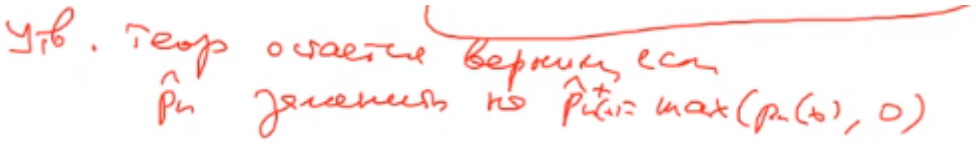
2. Посмотрим на ядра в более общем смысле: как функцию из $\mathbb{R}$ в $\mathbb{R}$.

   В чем смысл - не понял.

   Можно возразить: раньше была хорошая функция, а теперь у нас некрасивая, но более оптимальная. Но есть утверждение:



EM алгоритм

- единственный параметрический алгоритм оценки плотности, который хорошо работает

EM - excpectation maximization

Рассмотрим смесь двух нормальных распределений:

$$p(x) = (1-\pi)\, p_{(\mu_1, \sigma_1^2)}(x) + \pi\, p_{(\mu_2, \sigma_2^2)}(x)$$

Параметры и задача:

$$\vec{\theta} = (\mu_1, \sigma_1^2, \mu_2, \sigma_2^2, \pi)$$

Задача: оценить $\vec{\theta}$ по выборке $x_1, \dots, x_n$

Первое что приходит в голову - метод максимального правдоподобия

$$\text{ММП}: \quad \sum_{i=1}^{n} \log p(x_i) := \log L(x_1, \dots, x_n) \to \max_{\theta}$$

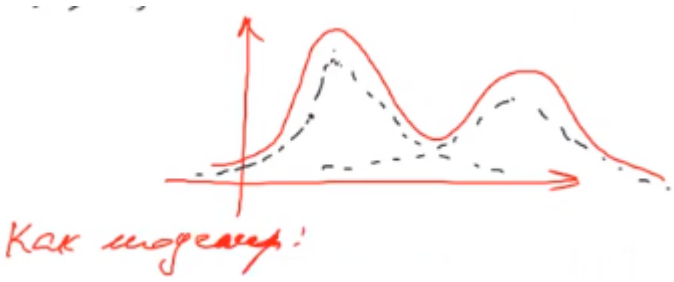
Это сложная оптимизационная задача. Можно использовать метода Бидоля-Равсона. Но это не очень.

Идея

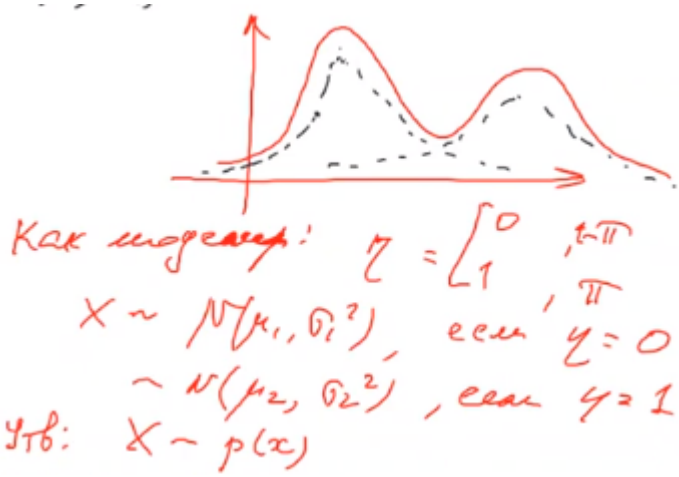
Введем фиктивные (латентные) переменные:

$$y_i = \begin{cases} 0, & x_i \sim N(\mu_1, \sigma_1^2) \\ 1, & x_i \sim N(\mu_2, \sigma_2^2) \end{cases}$$

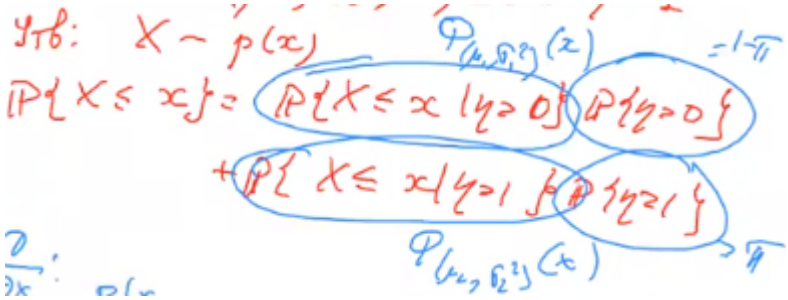
У нас распределение является смесью двух других



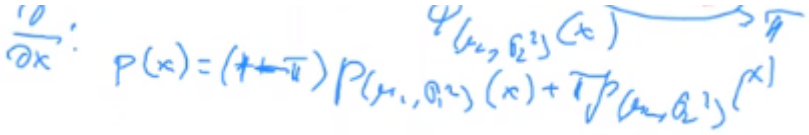
Отступление: как моделировать величину из смеси распределений



Как доказать? Считаем функцию распределения:

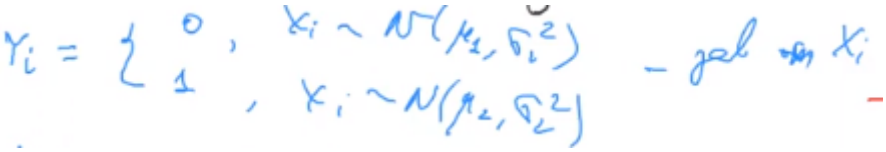


Дифференцируем:

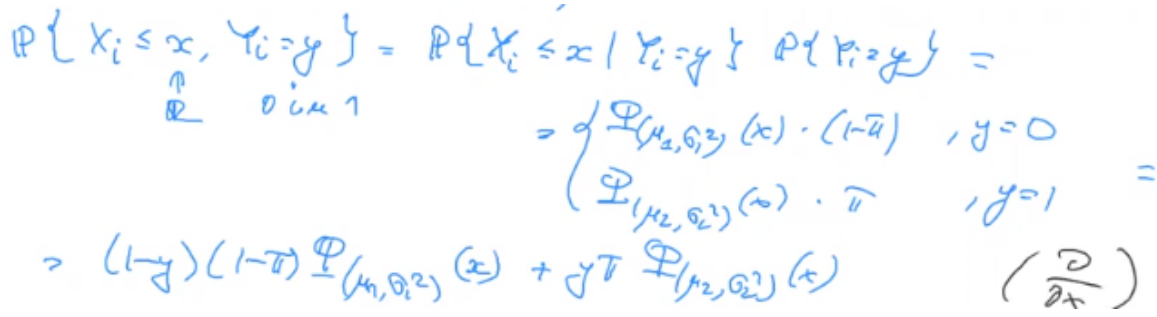


Возвращаемся

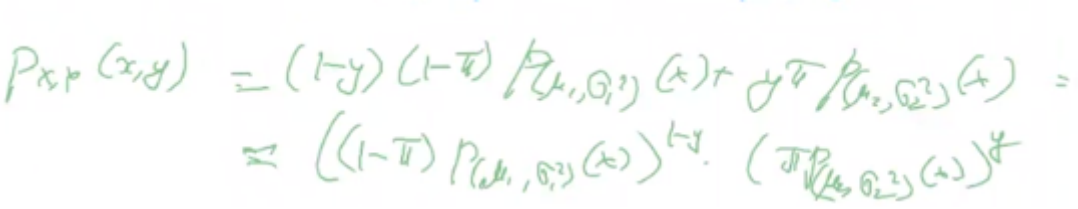
Итак, у нас есть Y_i:



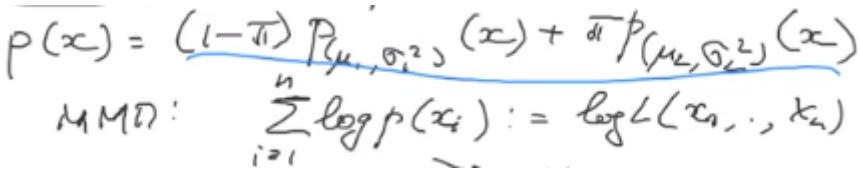
Тогда

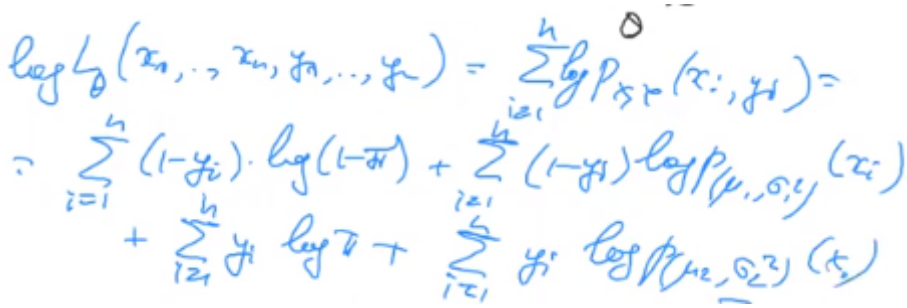


Дифференцируем и приводим к красивому виду:

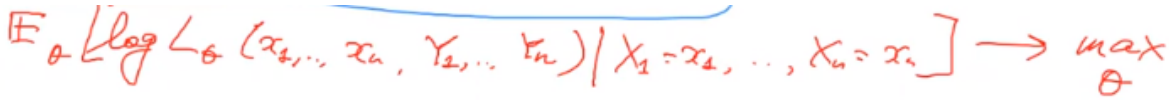


Записываем лог-правдободобие:



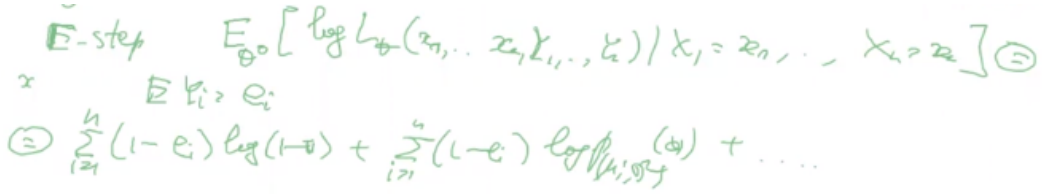


Но как отсюда вытащить оценку? Запишем $\log L_\theta(x_1, \dots x_n, y_1, \dots y_n)$ и вместо y_i подставим Y_i - случайные величины. А потом возьмем матожидание и максимизируем по θЭ

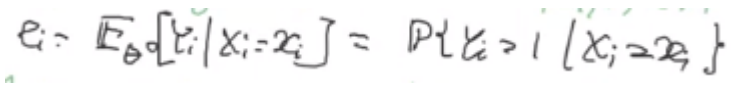


Как выглядит алгоритм

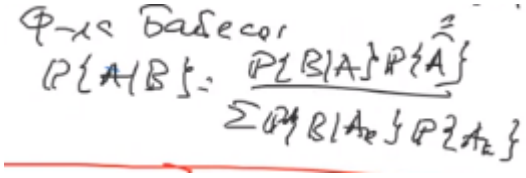
1. Выбираем $\theta^{(0)}$
2. $\mathbb{E}$-step



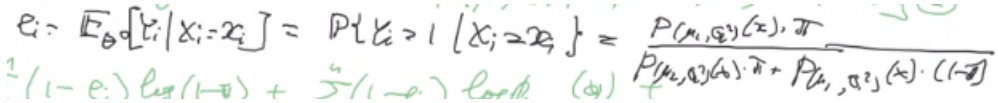
Тут



Вспомним формулу Байеса:



Тогда



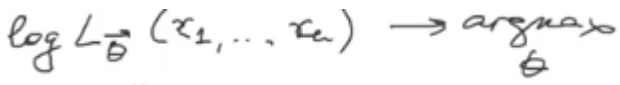
3. M-step:

4. Вернуться к шагу 1

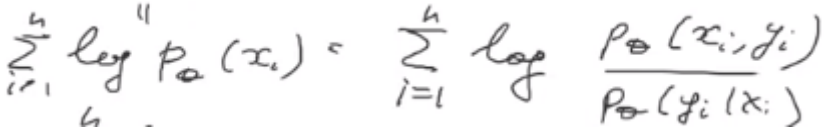
Зачем нужны были латентные переменные? Они постоянно убираются

Почему это работает?

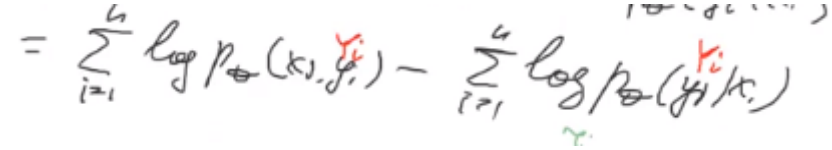
Изначальная идея - построить оценку лог правдободобия



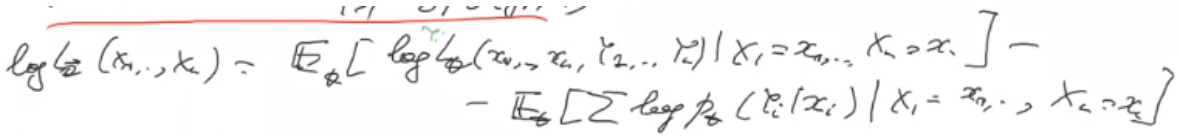
Преобразуем через условные вероятности:



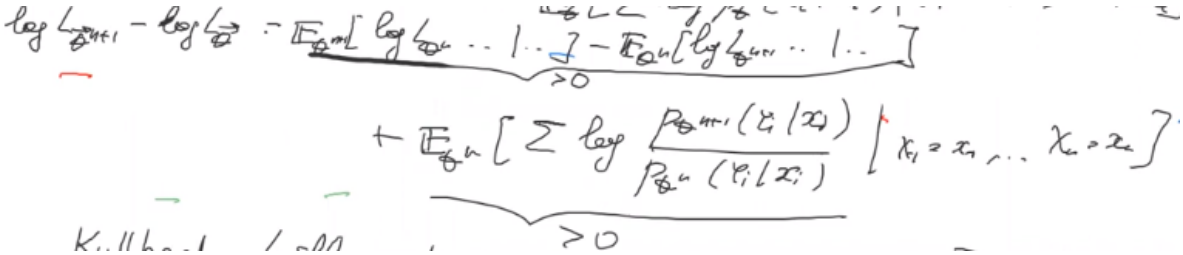
и распишем логорифм:



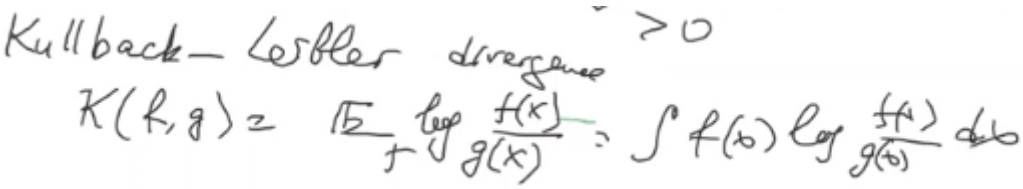
Т.к. y_i случайные, можно взять любые. Возьмем случайные y_i, то есть Y_i и используем матожидание:



Посмотрим, как у нас меняется логправдоподобие после итерации:



Второе слагаемое это KLD:



Получается, мы с каждый шагом работы EM алгоритма увеличиваем логорифм правдободия, а значит и приближаемся к ней. Возможно, мы попадем в локальный минимум, но в любом случае движемся мы в правильном направлении